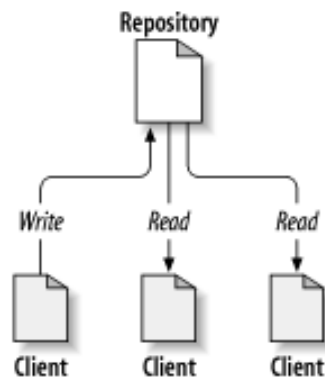


## Windows form & TortoiseSVN

ก่อนที่จะเรามาทวนในเรื่องของ Windows form เราจะมาทำความรู้จักกับ TortoiseSVN กันก่อนเพื่อให้เกิดความสะดวกในการทำโครงการ

## SVN

**SubVersion** นั้นเป็น tool อันหนึ่งในกลุ่มของ Version Control ครั้นนั้นคือเป็นตัวที่ควบคุม version ของ source code โดยที่เมื่อเราทำการแก้ไขแล้วเกิดปัญหาเราสามารถกลับไปเรียก version ที่ยังเป็นปกติกลับมาได้ โดยเริ่มต้นเราจะมาทำความเข้าใจ concept กับคำว่า repository กันก่อน โดยที่ repository นี้เปรียบเสมือนศูนย์กลางในการจัดเก็บข้อมูลที่ทุกคนใช้ร่วมกัน นั่นคือการ share ไฟล์ project เดียวกันให้ทุกคนที่ร่วมในการพัฒนาเมื่อมีการแก้ไขคนอื่น ๆ ก็จะสามารถ update และทำงานในส่วนของตนให้เข้ากับส่วนที่สมาชิกคนอื่นพัฒนาได้ในเวลาอันรวดเร็ว และยังเป็นการลดความสับสนในการคัดลอกไฟล์มารวมกันอีกด้วย



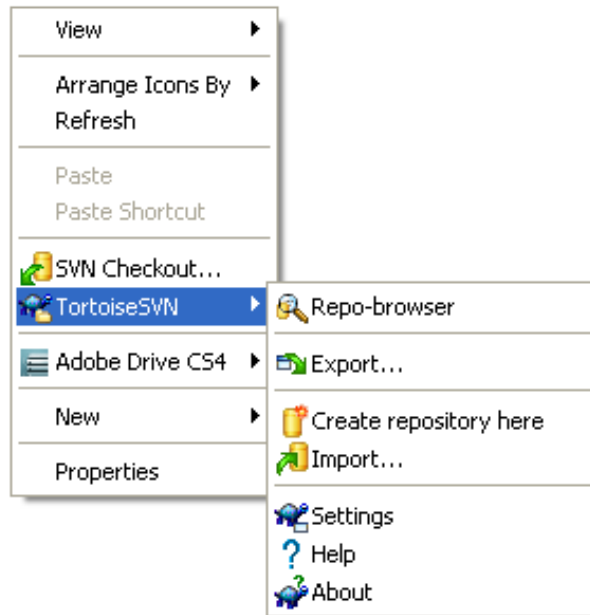
การเก็บ version จะเก็บในลักษณะ revision number โดยเมื่อมีการแก้ไขปรับปรุงในแต่ละครั้งจะทำให้ revision number เพิ่มขึ้น โดยที่ revision number ล่าสุดเราจะเรียกว่า Head revision

ระบบดังกล่าวจึงเป็นในลักษณะ server client โดยที่ server จะมี subversion<sup>1</sup> - server ให้บริการในการจัดการเก็บ source code ไว้ โดยที่ผู้ใช้งานจะใช้ SVN -client ในการทำงานซึ่งเราจะใช้ TortoiseSVN ในการทำ project ในครั้งนี้

<sup>1</sup> ในครั้งต่อไปจะขอใช้ คำว่า SVN แทน

## TortoiseSVN

TortoiseSVN หรือ เต่า เป็น SVN - client บน Windows ตัวหนึ่งที่มีความนิยมโดยทั่วไป สามารถดาวน์โหลดได้ที่ <http://tortoisesvn.tigris.org/> เมื่อติดตั้งแล้วเมื่อเรา คลิกขวาจะมีส่วนของ TortoiseSVN เพิ่มขึ้นมา



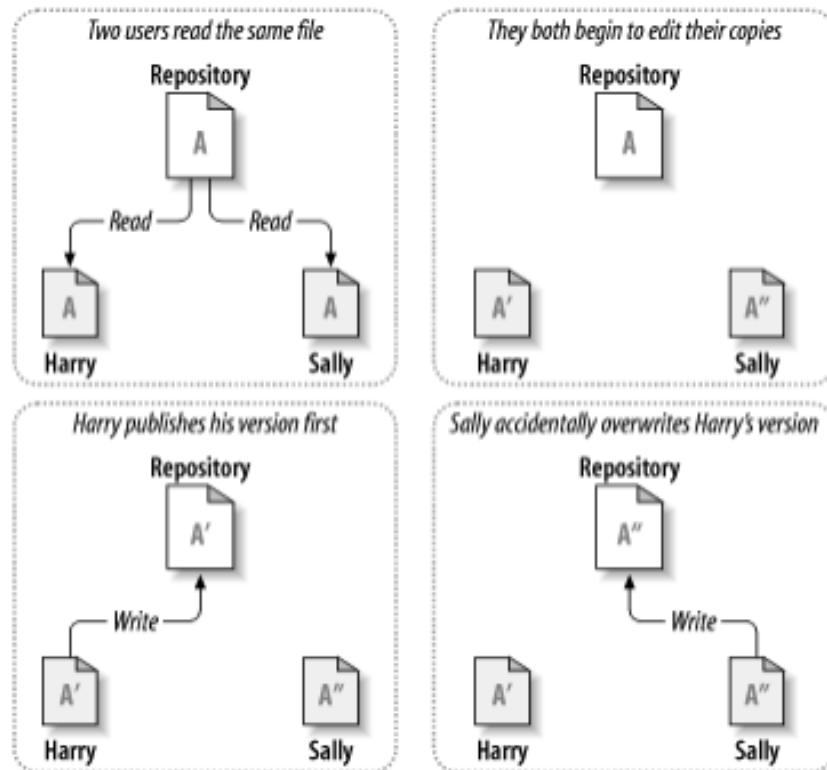
### คำสั่งพื้นฐานในการใช้งาน

- **Check out** ขั้นแรกเราต้องทำการดึงข้อมูลออกมาจาก repository ก่อน กระบวนการนี้เรียกว่า check out ซึ่งเราต้องระบุตำแหน่ง repository และ path ใน repository ที่เราต้องการ check out ออกมา หลังจาก check out ออกมาแล้ว ก็จะได้เป็น copy อยู่บนเครื่องของเรา อยากแก้ไขทำอย่างไรก็ได้  
อาจสงสัยว่า path ใน repository ทำไมเราต้องระบุด้วย นั้นเพราะใน repository หนึ่งๆ อาจมี project หลายๆ project ก็ได้ซึ่งเราอาจจะได้จัดการเพียงโปรเจกต์เดียว หรือ การทำงานใน project นั้นมีการจัดการแบบต้นไม้ (tree) ที่เราจะทำงานในส่วนเพียง กิ่งๆ (branch) หนึ่งเท่านั้น
- **Commit** หลัง check out ออกมาแล้ว หากเราทำการแก้ไขไฟล์บางไฟล์แล้วต้องการส่งกลับเข้าไปที่ repository เราสามารถทำได้โดยสั่ง commit สิ่งที่เราแก้ไขจะถูกส่งเข้าไปใน repository ทันที <sup>2</sup> หากมีคนอื่นมา check out ไปก็จะได้ไฟล์รุ่นที่เราแก้ไขแล้วไป

<sup>2</sup> หากไฟล์ดังกล่าวมีคนอื่น commit ขึ้นไปก่อน เราจะไม่สามารถ commit ต่อในทันทีเพราะเกิด conflict ขึ้น จะกล่าวในหัวข้อ conflict

- **Update** ถ้ามีคนอื่นที่แก้ไขไฟล์แล้ว commit กลับเข้าไปที่ repository เราก็จะยังไม่ได้รับการเปลี่ยนแปลงที่เกิดขึ้น ดังนั้นเราจึงต้องใช้คำสั่ง update
- **Add,Delete** เราจะเพิ่มไฟล์เข้าที่ repository ด้วยคำสั่ง add และลบออกจาก repository ด้วย delete

## Conflict



ลำดับการทำงานเป็นดังนี้

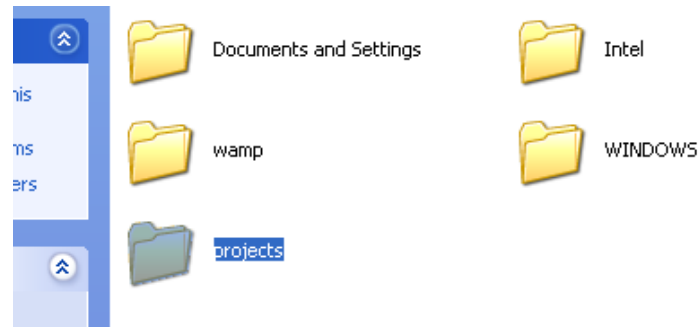
1. Harry check out A
2. Sally check out A
3. Harry edit A to A'
4. Sally edit A to A''
5. Harry commit A'
6. Sally commit A''

ขั้นตอนการทำงานดังกล่าวนี้จะไม่เกิดขึ้น เนื่องจากเกิด conflict ในขั้นตอนที่ 6 โดยจะเกิด out-of-date error ขึ้นซึ่งเราจะต้องจัดการแก้ conflict เองจะดีที่สุด (จะกล่าวทีหลัง)

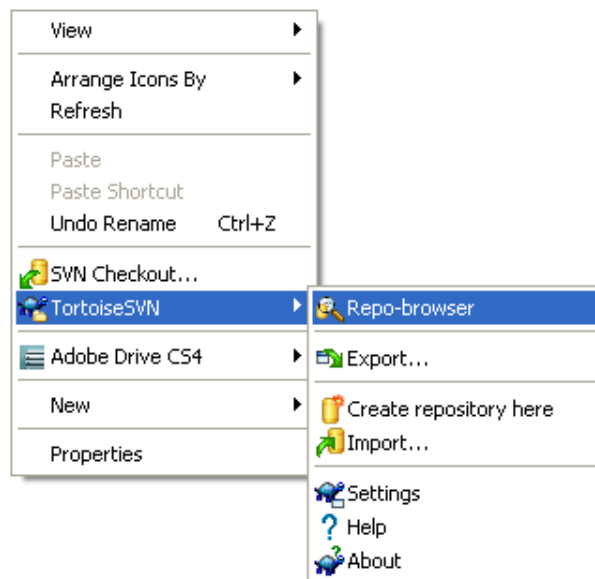
## FirstProject

ในหัวข้อนี้เราจะมาทดลองการใช้งาน TortoiseSVN กันโดยที่ repository ของแต่ละกลุ่มจะอยู่ที่ Url, username และ password ที่จะได้รับจากพี่เลี้ยง

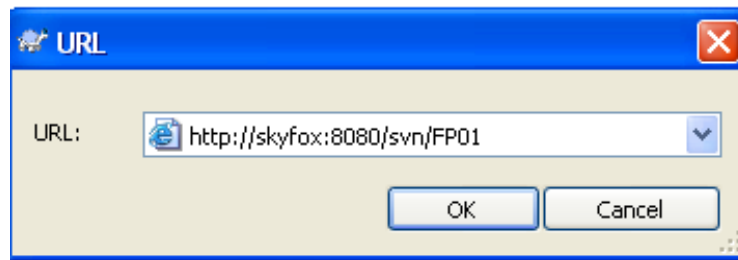
1.ให้เราสร้างที่เก็บ project ในเครื่องเราที่ c:\projects



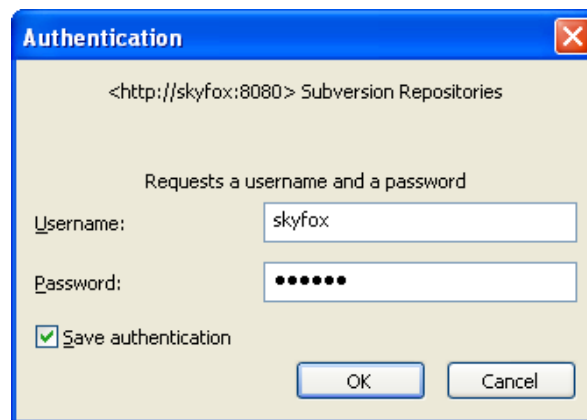
2.เราจะทำการสร้าง folder ใหม่สำหรับเก็บ project เราไว้ใน repository โดยคลิกขวาเลือก TortoiseSVN → Repo-browser



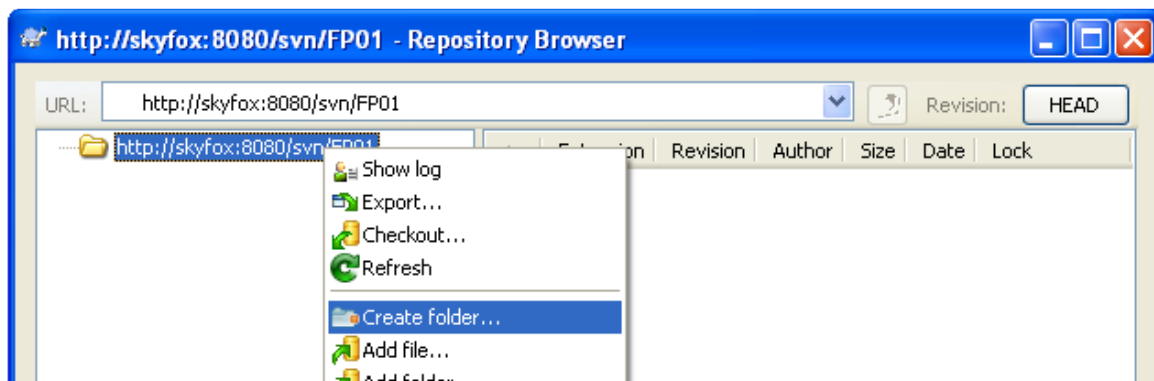
3. จะปรากฏหน้าต่าง URL ให้เราใส่ URL ของ repository ของเราลงไป  
(ในตัวอย่างอยู่ที่ `http://skyfox:8080/svn/FP01`)



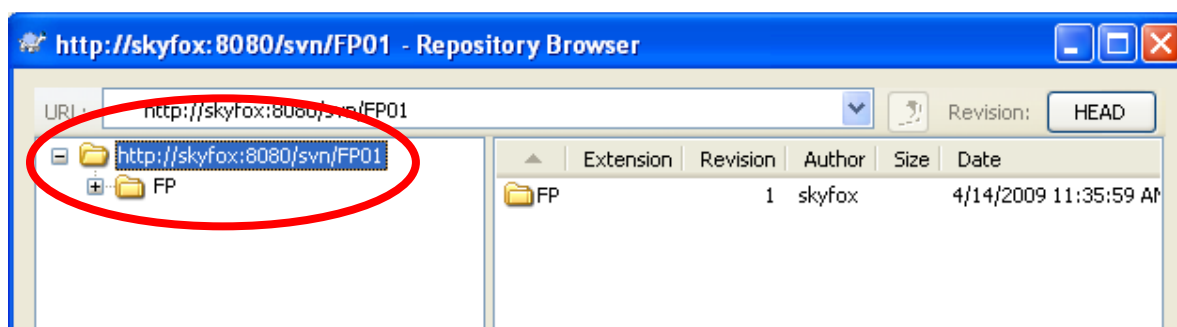
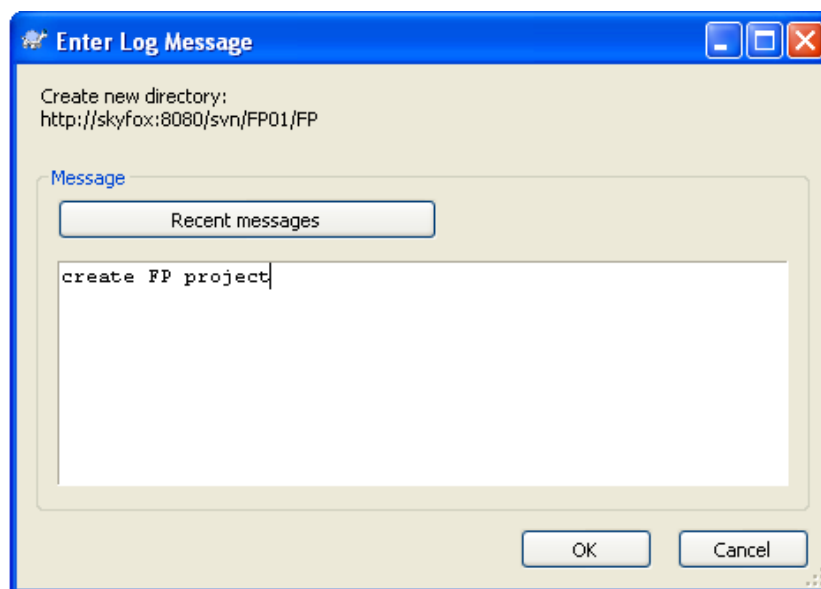
4. จะปรากฏหน้าต่าง Authentication เพื่อถาม username และ password  
(ให้เราเลือก save authentication ไปเลยเราจะได้ไม่ต้องมากรอกอีก)



5. ให้เราเลือกที่ root แล้ว สร้าง folder ใหม่ชื่อว่า FP



ในขณะที่สร้างจะปรากฏหน้าต่าง enter log message ให้เราใส่ข้อมูลที่เราทำอะไรลงไปเพื่อที่เวลาเรามาดูทีหลังจะได้รู้ว่า revision number นี้เราทำอะไรลงไปบ้าง

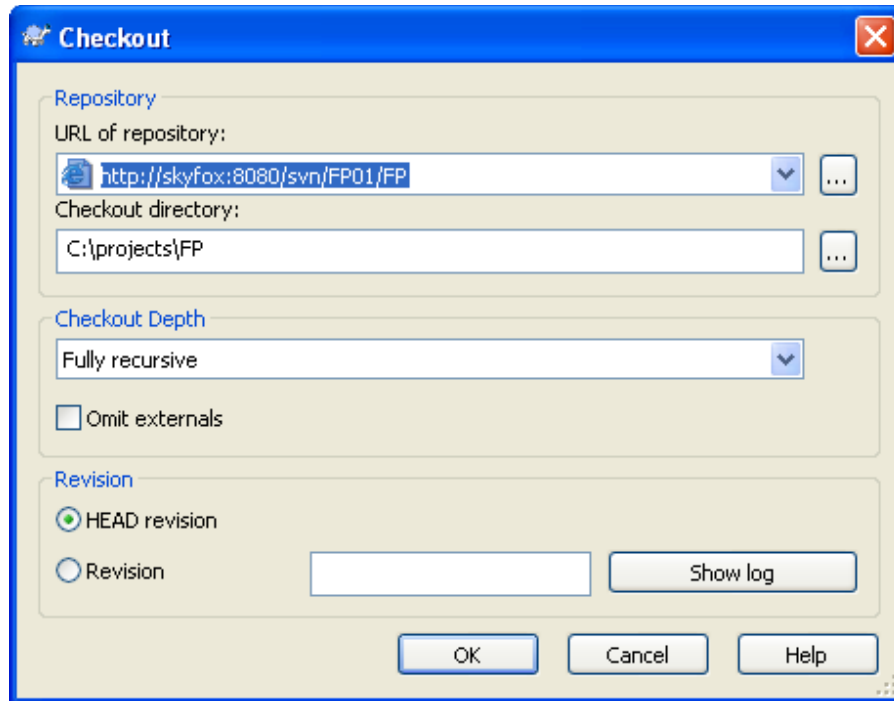


6. กลับมาที่ c:\projects ให้สร้าง folder FP แล้ว check out FP ออกมาจาก repository

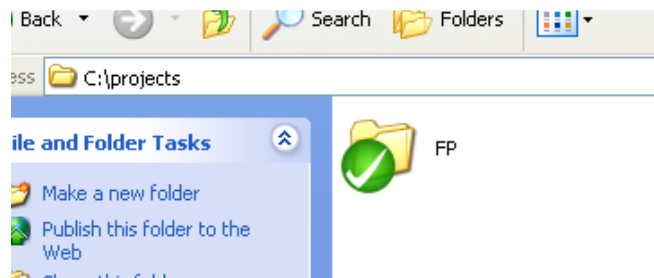
โดยคลิกขวา เลือก SVN Check out...

โดย URL เป็น <http://skyfox:8080/svn/FP01><sup>3</sup>/FP และ check out directory เป็น C:\projects\FP นั่นคือเป็นการ copy ไฟล์จาก <http://skyfox:8080/svn/FP01/FP> มาที่ C:\projects\FP

<sup>3</sup> ในตัวอย่างจะใช้ repository จากที่นี่



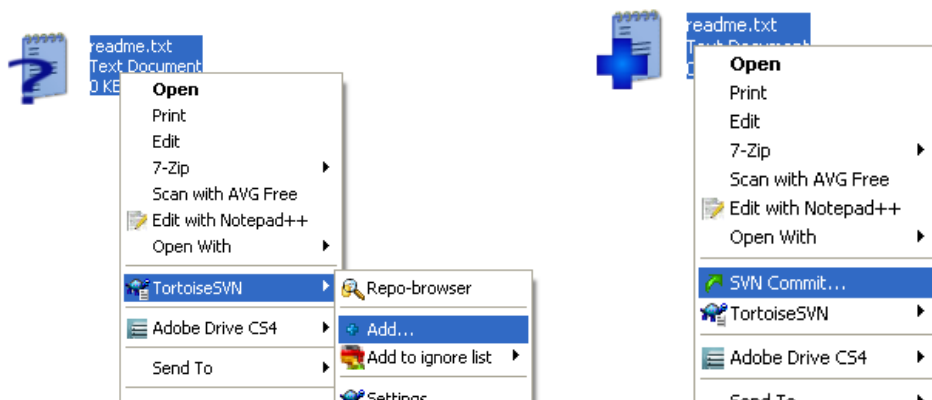
และที่ c:\projects จะปรากฏ folder FP แบบนี้ สัญลักษณ์ต่างๆดูเพิ่มเติมได้ที่ภาคผนวก

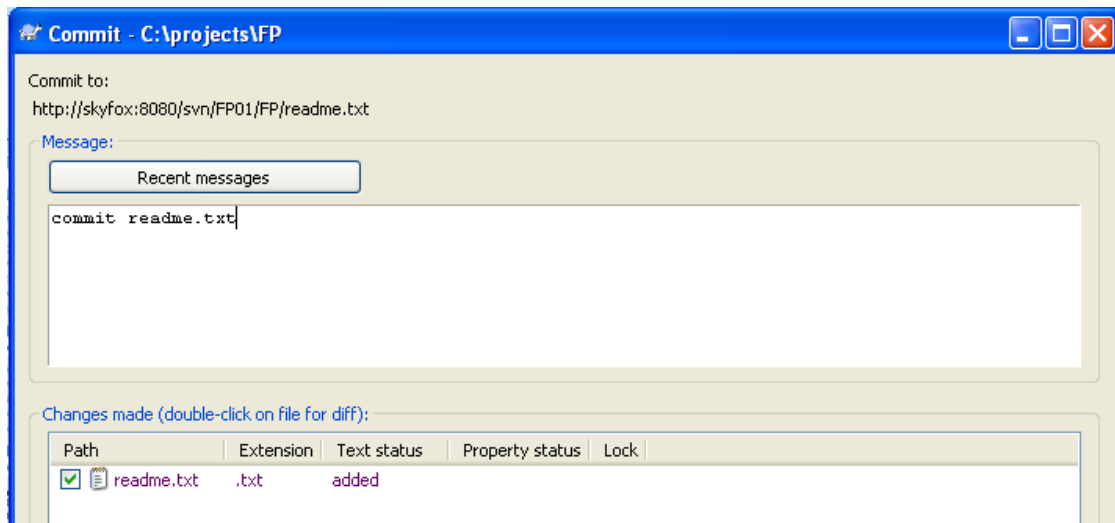


7. ให้เราสร้าง text document โดยที่คนแรกสร้าง readme.txt ส่วนอีกคนสร้าง help.txt ใน c:\projects\FP

แล้วทำการ Add โดยคลิกขวาที่ไฟล์ TortoiseSVN → Add

แล้ว commit ขึ้นไปที่ repository เช่นเดิมเราควรที่จะกรอก message เพื่อที่จะได้รู้ว่าเราทำอะไรไปใน revision นี้

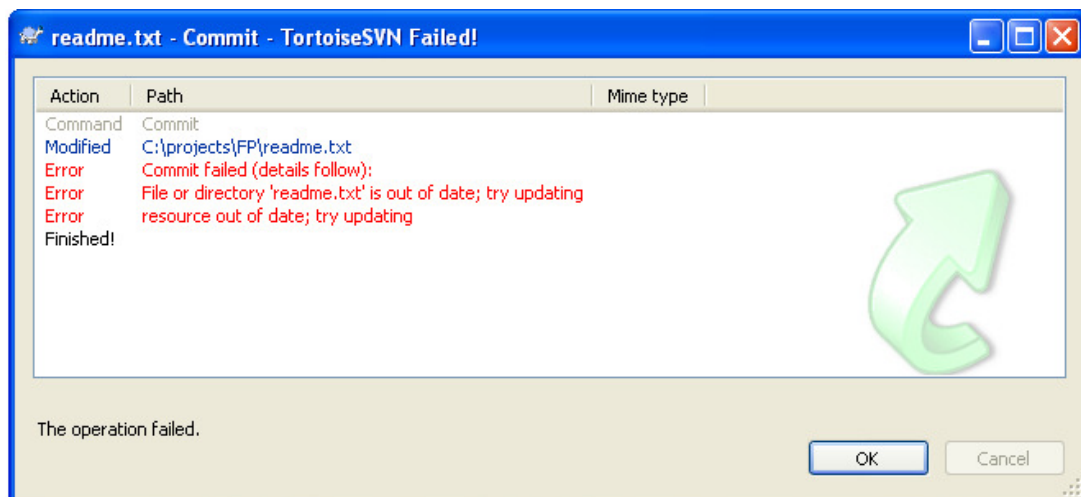




8. ให้แต่ละคนใช้คำสั่ง **Update** โดย คลิกขวา เลือก **SVN Update** เราจะได้ดังนี้

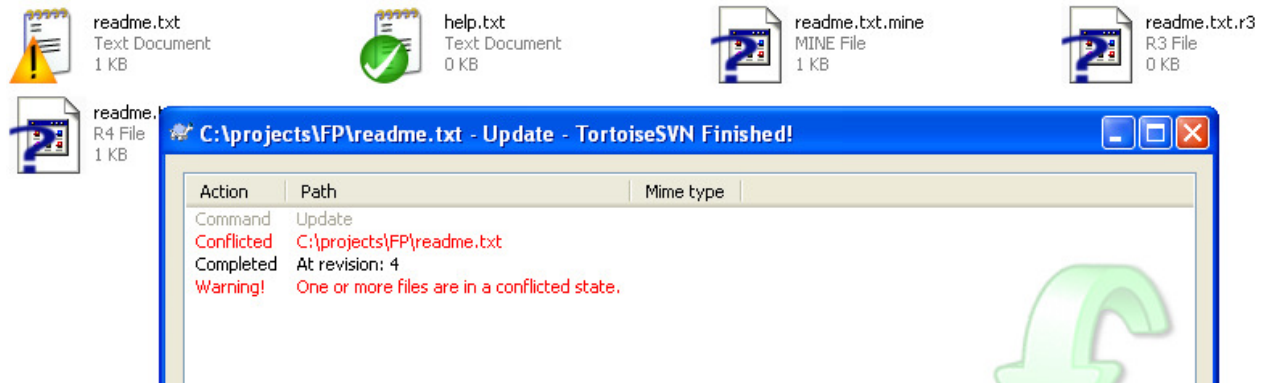


9. ให้แต่ละคนแก้ไข **readme.txt** save แล้ว **commit** ที่ละคน เมื่อคนแรก **commit** แล้วคนที่ 2 **commit** จะเกิด **conflict** แบบนี้

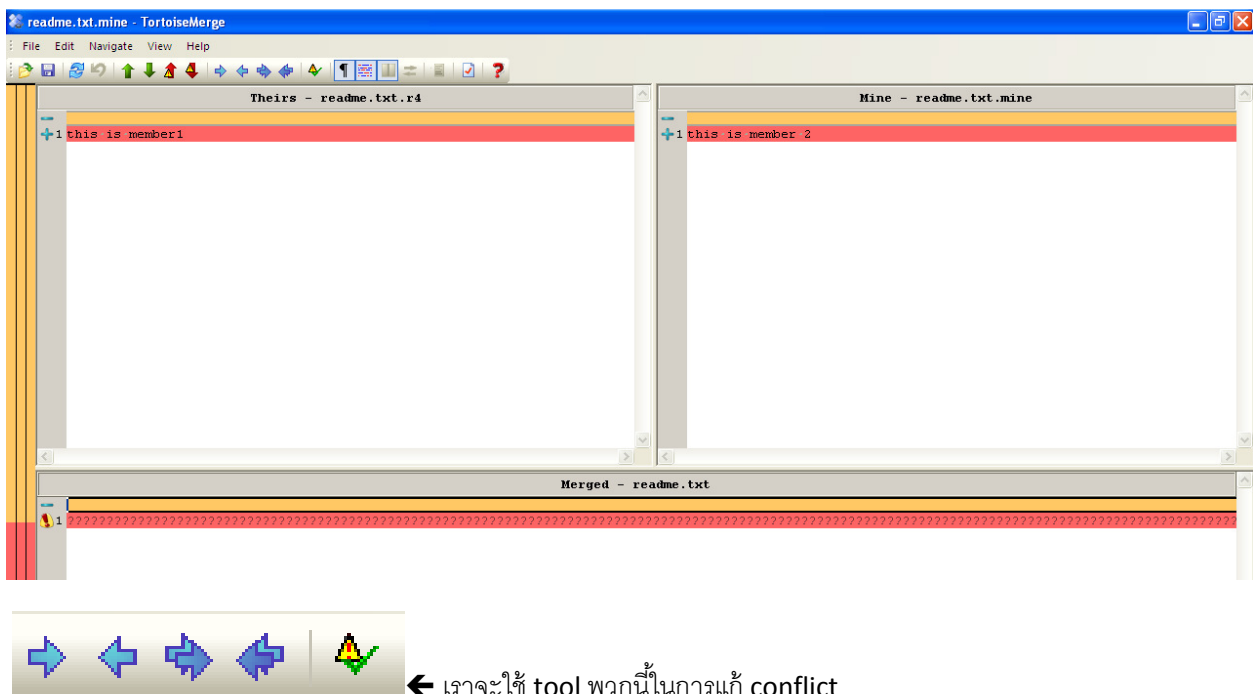


นั่นคือ คนที่ 2 จะต้อง **update** ก่อน เนื่องจากมีการแก้ไขไปแล้วพอ **update** ก็จะต้องมาแก้ **conflict** ที่เกิดอีก โดยที่จะเกิดไฟล์เพิ่มขึ้นมาดังรูป





เราจะแก้ conflict โดย คลิกขวาที่ไฟล์ที่มีเครื่องหมายอัศเจรีย์ TortoiseSVN → Edit conflicts จะมีหน้าต่างนี้เกิดขึ้น

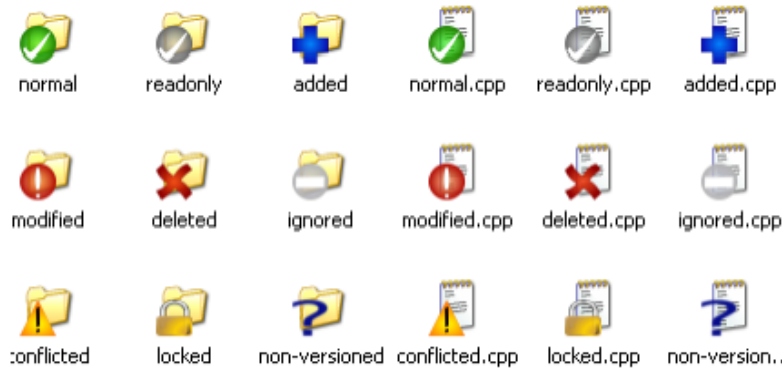


โดยในฝั่ง theirs จะเป็นฝั่งที่มาจาก repository และ mine เป็นของเรา โดยจุดที่เป็น conflict จะเป็น block สีแดงโดยให้เราคลิกที่จุดต่างๆแล้วใช้ tool ในการแก้ไข โดยข้อมูลที่แก้ไขแล้วจะปรากฏในช่องด้านล่าง

เมื่อเราแก้ไขครบแล้วให้คลิกที่ เครื่องหมาย Mark as resolved เราจะได้ไฟล์ที่แก้ไขแล้วเมื่อทดสอบโค้ดเรียบร้อยแล้วก็ให้ commit ขึ้นไปใหม่

## เพิ่มเติม

สัญลักษณ์ทั่วไปที่ปรากฏในการใช้งาน TortoiseSVN



### การใช้ revert

เมื่อเราเขียนโค้ดหรือได้แก้ไขไฟล์ไปแล้วอยากให้กลับไปเป็นอย่างเดิม คือต้องการลบที่แก้ไขทั้งหมดเราจะใช้ คำสั่ง **revert** หรือเราแก้ไขแล้วเกิด **conflict** เราต้องการ **update** ใหม่โดยที่เราต้องการลบการแก้ไขทั้งหมดของเราที่ทำไปเราก็จะใช้คำสั่งนี้เช่นกัน

### การใช้ ignore

เราจะใช้กับไฟล์ที่อยู่ใน **project** ที่เราไม่ได้ต้องการ **commit** ขึ้นไปที่ **repository** ของเรา เช่น ไฟล์ จำพวก exe หรือไฟล์คอนฟิกของเครื่องเราที่ไม่ได้ใช้กับคนอื่น ๆ

### การใช้ update to revision ...

เมื่อเราต้องการดึงโปรเจคไฟล์ออกมาโดยเราสามารถที่จะเลือกดึงออกมาตาม **revision number** ที่เราต้องการได้ โดยใช้คำสั่ง **update to revision ...** ซึ่งเราจะสามารถดู **version** ทั้งหมดได้จาก **log**

การใช้ SVN กับ project Windows application ของเรา

1. ให้ create project ของเราก่อนจากนั้นให้เราเข้าไปสร้าง folder project ของเราใน repository ตามขั้นตอนที่ 1 – 5 ที่เราได้ทำไปแล้วใน MyFirstProject
2. ให้เราไปที่ My Documents → Visual Studio 2008 → Projects
3. คลิกขวาที่ folder project ที่เราสร้างขึ้น เลือก SVN Checkout...
4. จะปรากฏหน้าต่างขึ้นมาให้เราใส่ดังนี้

**URL of repository**

`http://skyfox:8080/svn/group14/project_name5`

**Checkout directory**

`C:\Documents and Settings\Pop\My Documents6\Visual Studio 2008\Projects\project_name`

project\_name จะมีเพียงครั้งเดียวจะรับจะไม่เป็น \project\_name\project\_name

5. จะปรากฏหน้าต่างถาม ให้ตอบ Yes
6. ให้เราค่อยๆ add ไฟล์เข้าไปโดยไฟล์ที่เราจะ ignore มีดังนี้
  - ไฟล์ .csproject.user และไฟล์ .suo
  - folder bin
  - folder obj

เมื่อเราทำการ coding โปรแกรมเราเวลาที่จะ commit ให้เราตรวจดูให้ดีว่ามี ไฟล์ที่เรายังไม่ได้ add เพิ่มเข้าไปหรือไม่เช่นนั้นเมื่อ เรา update to revision ... กลับมาเราจะทำให้ไฟล์ที่เราไม่ได้ add ไม่มี ซึ่งจะทำให้ใช้งานไม่ได้

<sup>4</sup> url ของ repository พี่เลี้ยงจะเป็นคนบอก

<sup>5</sup> project\_name ในที่นี้คือ ชื่อ project ที่สร้างขึ้น

<sup>6</sup> ที่อยู่ของแต่ละคนอาจแตกต่างกัน

## WinApp (10 points + bonus) ทั้งหมดนี้มีตัวอย่างโปรแกรมให้โหลดมาดูเป็นตัวอย่างนะครับ

หลังจากเรียน SVN กันแล้วเรามาเขียนโปรแกรมประหลาดๆกันดีกว่า

### Level 1 (1 point)

ให้สร้างเครื่องคิดเลข บวก, ลบ, คูณ และหาร จำนวนเต็มโดยที่ output เป็น label ของเลขหลักหน่วยหลักเดียว และมี

MenuStrip ใช้ reset กับ exit

(Hint: this.Close();)

### Level 1.5 (3 points)

ให้มีปุ่ม countdown จาก output Lv1. มีปุ่ม pause เมื่อถึงจน 0 ให้ขึ้น Hello World แทนเลข 0

(Hint: interval = 1000)

### Level 2 (4 points)

ให้แก้จาก level 1.5 โดยให้ใช้รูปแทน label เป็นรูป 0 ถึง 9 และ Hello World

(Hint: import image เข้า pictureBox ,

```
pictureBox1.Image = global::WinApp.Properties.Resources._0; ใน
Form1.Designer.cs)
```

### Level 2.5 (2 points)

ให้มี radiobox ควบคุมให้เลขที่นับวิ่งจากซ้ายไปขวาหรือขวาไปซ้าย

(Hint: สร้าง timer อีกตัวให้ควบคุมการวิ่งในแนวราบ, ใช้คำสั่งด้านล่างกำหนดตำแหน่งของ PictureBox

```
pictureBox1.Location = new System.Drawing.Point(xlocation, ylocation);,
Interval = 10)
```

### Level 3 (bonus)

ให้สร้าง Progress bar แสดงความคืบหน้าการวิ่งของตัวเลข